

PROJETO DESENVOLVIDO NA DISCIPLINA ENG04009
U (14/2) INTRODUÇÃO À ENGENHARIA DE COMPUTAÇÃO
UFRGS

Projeto Final – Relatório Técnico Individual

Controle de Navegação EDUBOT

Rodrigo Dal Ri - 244936

25/11/2014

1. Objetivo

- Criar o algoritmo de controle de um robô móvel reativo com controle lógico, autônomo e que deve operar em ambientes desconhecidos.
- O algoritmo/firmware deve identificar obstáculos e verificar se executa ou não manobras evasivas, evitando a colisão.
- O objetivo é que o robô consiga sair deste labirinto sem intervenção humana.

2. Materiais utilizados

O projeto é composto de apenas três componentes de hardware: os motores DC, sonar, Arduino e uma matriz de contato ou protoboard com um driver Ponte – H para realizar a conexão dos motores.

O arduino serve como ponte entre os sensores (sonar) e os atuadores (motores), transmitindo a eles as instruções provenientes do firmware/software.

Elementos Mecânicos:

- Rodas de tração independentes;
- Rodas de apoio;
- Chassi e Carenagem;

Elementos Eletromecânicos:

- Motor de corrente contínua;
- Sensor de Ultrassom;

Elementos Computacionais:

- Arduino Duemilanove;
- Driver SN754410 – “Ponte H”;

3. Soluções Propostas

Estratégia de navegação:

- A estratégia de navegação é bem simples. O robô tem como movimento preferencial dobrar a direita, depois ir para esquerda e por último ir a frente.
- Então sempre que o sensor verifica que não há obstáculos a direita ele faz a curva, se houver obstáculos ele verifica se existem obstáculos a esquerda. Se não houver ele anda, caso contrário ele testa se existem obstáculos a frente. Se não houver ele anda para a frente.

Código:

```
/*
MOD0 pinol pino2
morto 0 0
horario 1 0
anterior 0 1
freio 1 1
*/
//diatancia minima parada = 9
//delay para virar 90° = 650

#include <Servo.h>
#define DIR_PIN1 3
#define DIR_PIN2 4
#define DIR_VEL 9
#define ESQ_PIN1 5
#define ESQ_PIN2 6
#define ESQ_VEL 10
#define DIR 15
#define FRE 90
#define ECHO 12
#define TRIG 13
//potência do motor
#define XABLAU 255

unsigned int d;
int pos=0;
Servo myservo;
int i=0;

void setup()
{
    setupUltrasonic();
    Serial.begin(9600);
    myservo.attach(7);
    myservo.write(90);
    setupMove();
}

void setupMove()
{
    pinMode(ESQ_PIN1, OUTPUT);
    pinMode(ESQ_PIN2, OUTPUT);
    pinMode(ESQ_VEL, OUTPUT);
    pinMode(DIR_PIN1, OUTPUT);
    pinMode(DIR_PIN2, OUTPUT);
    pinMode(DIR_VEL, OUTPUT);
    analogWrite(ESQ_VEL, XABLAU);
    analogWrite(DIR_VEL, XABLAU);
}

void setupUltrasonic()
{
    pinMode(TRIG, OUTPUT);
    pinMode(ECHO, INPUT);
}
```

```

void moveTras()
{
  digitalWrite(DIR_PIN1, LOW);
  digitalWrite(DIR_PIN2, HIGH);
  digitalWrite(ESQ_PIN1, LOW);
  digitalWrite(ESQ_PIN2, HIGH);
}
void moveFrente()
{
  digitalWrite(DIR_PIN1, HIGH);
  digitalWrite(DIR_PIN2, LOW);
  digitalWrite(ESQ_PIN1, HIGH);
  digitalWrite(ESQ_PIN2, LOW);
}
void moveDireita()
{
  digitalWrite(DIR_PIN1, LOW);
  digitalWrite(DIR_PIN2, LOW);
  digitalWrite(ESQ_PIN1, HIGH);
  digitalWrite(ESQ_PIN2, LOW);
}
void moveEsquerda()
{
  digitalWrite(DIR_PIN1, HIGH);
  digitalWrite(DIR_PIN2, LOW);
  digitalWrite(ESQ_PIN1, LOW);
  digitalWrite(ESQ_PIN2, HIGH);
}
void movePara()
{
  digitalWrite(DIR_PIN1, LOW);
  digitalWrite(DIR_PIN2, LOW);
  digitalWrite(ESQ_PIN1, LOW);
  digitalWrite(ESQ_PIN2, LOW);
}

unsigned int medeDist()
{
  int distancia;
  digitalWrite(TRIG, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG, LOW);
  distancia = pulseIn(ECHO, HIGH);
  return distancia/58;
}
void giraCabecote()
{
  for(pos = 0; pos <= 180; pos += 1){
    myservo.write(pos);
    delay(15);}
  for(pos = 180; pos>=0; pos-=1){
    myservo.write(pos);
    delay(15);}
}

```

```

void curvaDireita()
{
    moveFrente();
    delay(200);
    moveDireita();
    delay(650);
    moveFrente();
    delay(450);
    movePara();
    delay(200);
}

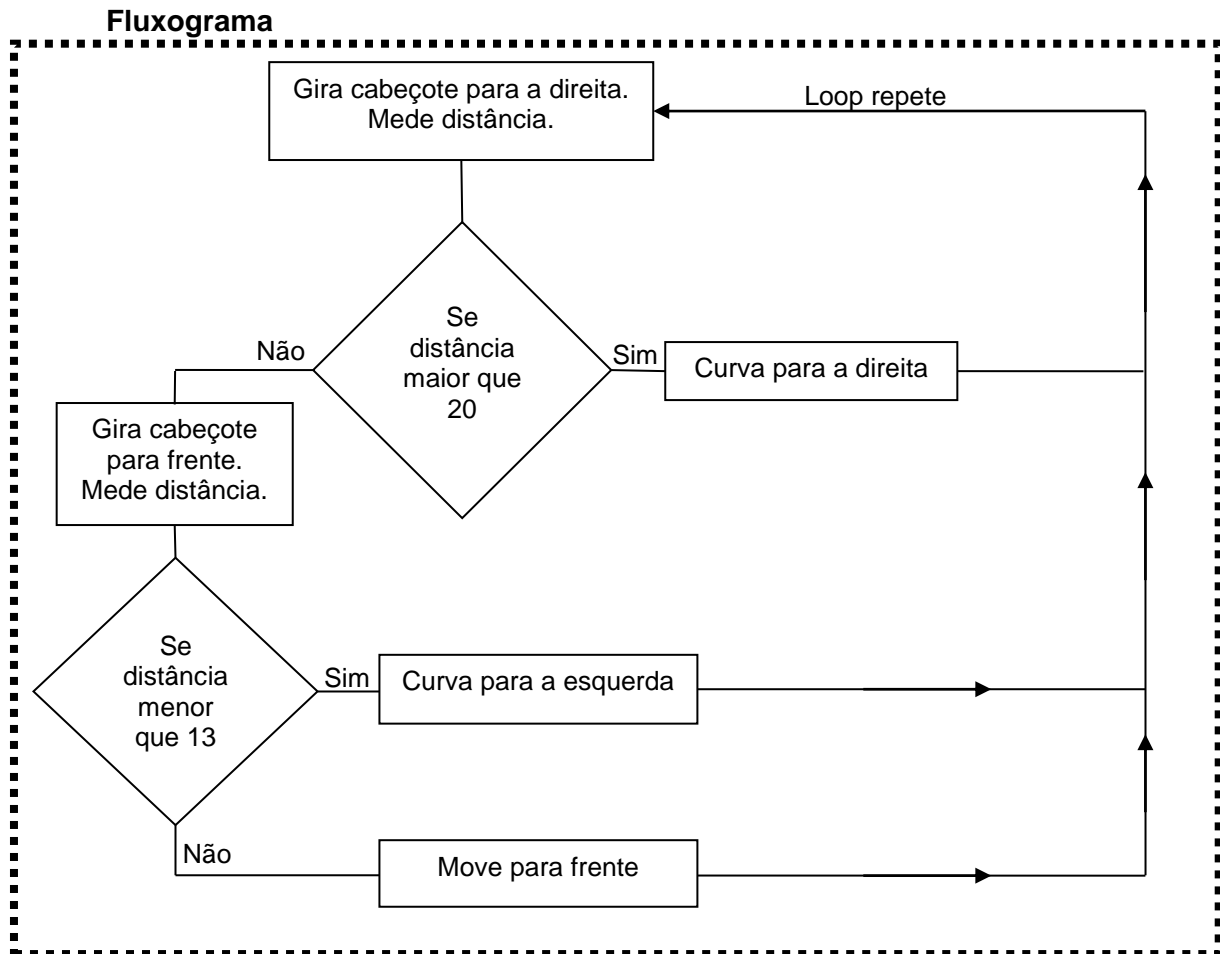
void curvaEsquerda()
{
    unsigned int menor=40;
    moveEsquerda();
    delay(200);
    movePara();
    myservo.write(DIR);
    delay(70);
    d=40;
    do{
        menor=d;
        d = medeDist();
        moveEsquerda();
        delay(50);
        movePara();
        delay(100);
    }while(d<menor);
    movePara();
    delay(200);
}

```

```

void loop()
{
    movePara();
    myservo.write(DIR);
    delay(400);
    d = medeDist();
    if(d>20)
        curvaDireita();
    else{
        myservo.write(FRE);
        delay(300);
        d = medeDist();
        if(d<13)
            curvaEsquerda();
        else{
            moveFrente();
            delay(400);
            moveDireita();
            delay(25);
            movePara();
            delay(100);}}
}

```



4. Resultados obtidos

- Nos primeiros testes implementamos apenas as funções para moverem o robô nas quatros direções, todas funcionaram sem problemas.
- Nos testes seguintes calibramos o servo e sensor ultrassom, para testar o ultrassom usamos o serial monitor da IDE do arduino.
- No teste final, colocamos o robô no labirinto e verificamos que ele cumpriu seus objetivos.

5. Análise Crítica

O que foi aprendido e os principais resultados alcançados:

- Aprendi a codificar na linguagem do arduino, sobre o funcionamento de micro controladores, sensores, no caso ultrassom e atuadores.
- Desenvolvi o raciocínio lógico na busca de um algoritmo eficiente para o robô sair do labirinto.
- E também melhorei minhas habilidades de trabalho em grupo, como por exemplo trocas de conhecimentos.

Principais obstáculos ou dificuldades encontradas:

- O principal obstáculo enfrentado foi o de pensar em uma lógica que fizesse o robô sair do labirinto sem a intervenção humana.